

УДК 004.912

Вікторія Костенко, Владислав Соколенко, Владислав Сасенко, Дмитро Оболкін
Університет митної справи та фінансів, Україна

ПРОБЛЕМИ РОЗРОБКИ ІНФОРМАЦІЙНИХ СИСТЕМ

Показані деякі проблеми, які виникають під час проектування та розробки інформаційних систем. Розглядаються переваги та недоліки при використанні методології USDP/RUP, а також деякі моменти, які б дозволили проводити автоматичне узгодження компонентів в створюваних інформаційних системах.

Ключові слова: інформаційна система, дані, ітеративно-інкрементальна модель розробки ІС, методологія USDP/RUP.

Viktoriia Kostenko, Vladyslav Sokolenko, Vladyslav Sayenko, Dmytro Obolkin
PROBLEMS OF DEVELOPMENT OF INFORMATION SYSTEMS

There are considered some problems which arise during the design and development of information systems. Discuss the advantages and disadvantages of using the USDP/RUP methodology, as well as some points that would allow automatic matching of components in the created information systems.

Keywords: information system, data, iterative-incremental model of IS development, USDP/RUP methodology.

Терміном "інформаційне суспільство" сьогодні досить широко користуються для означення деяких визначальних для людського життя змін у сучасному суспільстві, що спричиняються інтенсифікацією в ньому інформаційних потоків на базі розвитку інформаційно-комп'ютерних технологій.

У зв'язку з широкою комп'ютеризацією суспільства на сучасному етапі пред'являються високі вимоги до апаратної частини комп'ютерів, програмного забезпечення та інформаційних систем. Створення і функціонування інформаційної системи – складний і багатоетапний процес, який вимагає чіткої структуризації робіт і певної методології впровадження нових засобів.

У зв'язку з цим пропонується використання поняття життєвого циклу інформаційної системи, який являє собою безперервний процес її побудови та розвитку технічного завдання аж до вилучення з експлуатації.

Кожен етап життєвого циклу включає в себе певний склад, послідовність здійснюваних робіт і їх безпосередні результати. Окремий етап характеризується різними методами і засобами, використовуваними для виконання робіт, а також різними ролями і відповідальністю учасників. Результатом такого детального опису етапів життєвого циклу служить чітко спланований і організований процес колективної розробки інформаційної системи.

Повний життєвий цикл інформаційної системи включає в себе, як правило, стратегічне планування, аналіз, проектування, реалізацію, впровадження та експлуатацію. При цьому життєвий цикл, в свою чергу, може ділитися на стадії.

Але це тягне за собою одну з неприємних проблем – ускладнення (у т.ч. – уповільнення) повноцінного функціонування.

А якщо ж взагалі, то цей поділ є досить варіативним і часто довільним.

На сучасному етапі найбільш поширеними моделями життєвого циклу інформаційних систем є наступні:

- каскадна;
- спіральна;
- інкрементна.

У інкрементній моделі ЖЦ ІС розробка інформаційної системи здійснюється з лінійної послідовності стадій, але в кілька інкрементів (версій), тобто з запланованим поліпшенням продукту.

При використанні тієї чи іншої моделі важливо зрозуміти, якою мірою вона може бути корисна для тих, хто реально зайнятий експлуатацією інформаційних систем.

Але, використання лише однієї моделі для розробки ІС говорить про «монополізм» (якщо так можна говорити) та однобічність у цій сфері. Використання одних і тих же методів розробки веде до відсутності новизни.

На протязі вже багатьох років спостерігається цікава (і не дуже правильна) тенденція. Полягає вона у тому, що одним з основних способів побудови ІС є так званий спосіб розробки виключно «під себе». Мабуть, пов'язано це з тим, що існують лише загальні принципи та правила побудови ІС. Це є проблемою, адже за такого принципу розробки одну систему буде дуже складно інтегрувати в іншу. А якщо при цьому використовується лише так званий прототип (коли замість повноцінно функціонуючої системи створюється деяка обмежена версія, яка відповідає основним потребам користувачів), то про поняття інтегрування можна взагалі забути. І тим складніше буде розвивати таку систему.

Недосконало або обмежено спроектовані та розроблені системи нівелюють одну з найважливіших особливостей – повноту і достатність циркулюючих в ІС даних. Це призводить до наступних проблем:

- функцій управління не реалізовані до кінця;
- інформації не надається своєчасно;
- порушується достовірність інформації;
- витрати на обробку даних перевищують одержуваний ефект;
- неадаптованість до мінливих інформаційних потреб користувачів.

В основі методології USDP/RUP лежить ідея розробки ПЗ як процесу послідовного створення різних моделей, що рухаються від предметної області і вимог до програмного коду системи – до формальної специфікації ПЗ у вигляді виконуваного коду.

Чим ця методологія є цікавою?

По-перше, процес розробки управляється випадками використання (use cases) створюваної системою – тобто відстеженням тієї функціональності, яка потрібна, значима для користувачів.

По-друге, присутня орієнтація на архітектуру системи.

По-третє, присутня так звана ітеративно-інкрементальна модель розробки – вимоги, архітектура, програмні компоненти та інші артефакти розробляються не за один годину або день, а постійно уточнюються, в результаті чого виявляються нові вимоги до них, нові властивості, може відбуватися їх реструктуризація.

Таким чином відбувається подолання проблем, пов'язаних з складністю та перевантаженістю, а також мінливістю ПЗ.

Є ще одна річ, про яку не треба забувати. Це складність при розробці потреб інформаційної системи. Це дійсно складно. Тому що більшість програмних систем унікальні (індивідуальності клієнтів і компаній-замовників, своєрідність бізнес-ситуації, різні варіації технологій і т.д.).

Що цікаво, у ряді випадків розробники ПЗ, як правило, не є абсолютними фахівцями в тій предметній області, для якої призначене їх ПЗ. А інколи – користувачі і замовник не знають толком, яка система їм потрібна.

Сфера діяльності замовника (де повинна з'явитися замовлена система) може змінюватися. А отже, вимоги схильні змінюватися, поки система розробляється.

Звісно, що створити повне автоматичне узгодження різних візуальних моделей та компонентів в ІС, яка розробляється, неможливо. Занадто великі семантичні та онтологічні “прірви” між ними. Якась доля автоматизації тут, безумовно, можлива. Але це регламентовано тим, що при створенні стійкої моделі вимог і стабільної архітектури ІС всі моделі повинні перероблятися, і, схоже, що не по одному разу (щоб залишитися актуальними).

Потім це принесе свої результати. Наприклад, при внесенні незначних змін в проект, можна легко простежити, що і де потрібно змінити. Але до повного автоматизму ще дуже далеко.

Література

1. Анісімов В.В. Проектування інформаційних систем / В.В. Анісімов – Х.: Вид-тво ДВГУПС, 2006. – 112 с.
2. Кознов Д. В. Основы визуального моделирования / Д.В. Кознов. – М: БИНОМ, 2007. – 248 с.